(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
1 March 2001 (01.03.2001)

PCT

(10) International Publication Number
WO 01/14991 A2

(51) International Patent Classification[7]: G06F 15/16

(21) International Application Number: PCT/US00/23349

(22) International Filing Date: 24 August 2000 (24.08.2000)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/150,453 24 August 1999 (24.08.1999) US
09/383,340 25 August 1999 (25.08.1999) US

(71) Applicant: NETWORK APPLIANCE, INC. [US/US]; 495 East Java Drive, Sunnyvale, CA 94089 (US).

(72) Inventor: KLEIMAN, Steven, Robert; 157 El Monte Court, Los Altos, CA 94022 (US).

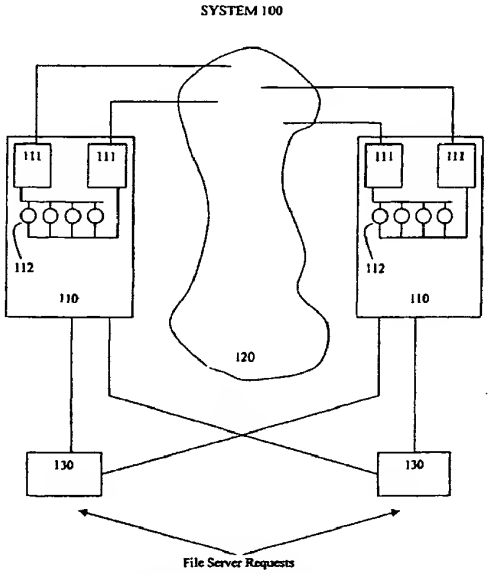
(74) Agent: SWERNOFSKY, Steven, A.; Swernofsky Law Group, P.O. Box 390013, Mountain View, CA 94039-0013 (US).

(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

[Continued on next page]

(54) Title: SCALABLE FILE SERVER WITH HIGHLY AVAILABLE PAIRS



(57) Abstract: The invention provides a file server system and a method for operating that system, which is easily scalable in number and type of individual components. A plurality of file servers are coupled using inter-node connectivity, such as an inter-node network, so that any one node can be accessed from any other node. Each file server includes a pair of file server nodes, each of which has a memory and each of which conducts file server operations by simultaneously writing to its own memory and to that of its twin, the pair being used to simultaneously control a set of storage elements such as disk drives. File server requests directed to particular mass storage elements are routed among file servers using an inter-node switch and processed by the file servers controlling those particular storage elements. The mass storage elements are disposed and controlled to form a redundant array, such as a RAID

[Continued on next page]

WO 01/14991 A2

Published:
— *Without international search report and to be republished upon receipt of that report.*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

storage system. The inter-node network and inter-node switch are redundant, so that no single point of failure prevents access to any individual storage element. The file servers are disposed and controlled to recognize failure of any single element in the file server system and to provide access to all mass storage elements despite any such failures.

SCALABLE FILE SERVER WITH HIGHLY AVAILABLE PAIRS

Background of the Invention

1. *Field of the Invention*

The invention relates to storage systems.

2. *Related Art*

Computer storage systems are used to record and retrieve data. One way storage systems are characterized is by the amount of storage capacity they have. The capacity for storage systems has increased greatly over time. One problem in the known art is the difficulty of planning ahead for desired increases in storage capacity. A related problem in the known art is the difficulty in providing scalable storage at a relatively efficient cost. This has subjected customers to a dilemma; one can either purchase a file system with a single large file server, or purchase a file system with a number of smaller file servers.

The single-server option has several drawbacks. (1) The customer must buy a larger file system than currently desired, so as to have room available for future expansion. (2) The entire file system can become unavailable if the file server fails for any reason. (3) The file system, although initially larger, is not easily scalable if the customer comes to desire a system that is larger than originally planned capacity.

The multi-server option also has several drawbacks. In systems in which the individual components of the multi-server device are tightly coordinated, (1) the same scalability problem occurs for the coordinating capacity for the individual components. That is, the customer must buy more coordinating capacity than currently desired, so as to have room available for future expansion. (2) The individual components are themselves often obsolete by the time the planned-for

greater capacity is actually needed. (3) Tightly coordinated systems are often very expensive relative to the amount of scalability desired.

In systems in which the individual components of the multi-server device are only loosely coordinated, it is difficult to cause the individual components to behave in a coordinated manner so as to emulate a single file server. Although failure of a single file server does not cause the entire file system to become unavailable, it does cause any files stored on that particular file server to become unavailable. If those files were critical to operation of the system, or some subsystem thereof, the applicable system or subsystem will be unavailable as a result. Administrative difficulties generally increase to due to a larger number of smaller file servers.

Accordingly, it would be advantageous to provide a method and system for performing a file server system that is scalable, that is, which can be increased in capacity without major system alterations, and which is relatively cost efficient with regard to that scalability. This advantage is achieved in an embodiment of the invention in which a plurality of file server nodes (each a pair of file servers) are interconnected. Each file server node has a pair of controllers for simultaneously controlling a set of storage elements such as disk drives. File server commands are routed among file server nodes to the file server node having control of applicable storage elements, and in which each pair of file servers is reliable due to redundancy.

It would also be advantageous to provide a storage system that is resistant to failures of individual system elements, and that can continue to operate after any single point of failure. This advantage is achieved in an embodiment of the invention like that described in International Application PCT/US99/05071, filed 8 March 1999, in the name of the same applicant, titled "Highly Available File Servers," hereby incorporated by reference as if fully set forth herein.

Summary of the Invention

The invention provides a file server system and a method for operating that system, which is easily scalable in number and type of individual components. A plurality of file server nodes (each a pair of file servers) are coupled using inter-node connectivity, such as an inter-node network, so that any one pair can be accessed from any other pair. Each file server node includes a pair of file servers, each of which has a memory and each of which conducts file server operations by simultaneously writing to its own memory and to that of its twin, the pair being used to simultaneously control a set of storage elements such as disk drives. File server commands or requests directed to particular mass storage elements are routed among file server nodes using an inter-node switch and processed by the file server nodes controlling those particular storage elements. Each file server node (that is, each pair of file servers) is reliable due to its own redundancy.

In a preferred embodiment, the mass storage elements are disposed and controlled to form a redundant array, such as a RAID (Redundant Array of Independent Disks) storage system. The inter-node network and inter-node switch are redundant, and file server commands or requests arriving at the network of pairs are coupled using the network and the switch to the appropriate pair and processed at that pair. Thus, each pair can be reached from each other pair, and no single point of failure prevents access to any individual storage element. The file servers are disposed and controlled to recognize failures of any single element in the file server system and to provide access to all mass storage elements despite any such failures.

Brief Description of the Drawings

Figure 1 shows a block diagram of a scalable and highly available file server system.

Figure 2A shows a block diagram of a first interconnect system for the file server system.

Figure 2B shows a block diagram of a second interconnect system for the file server system.

Figure 3 shows a process flow diagram of operation of the file server system.

Detailed Description of the Preferred Embodiment

In the following description, a preferred embodiment of the invention is described with regard to preferred process steps and data structures. However, those skilled in the art would recognize, after perusal of this application, that embodiments of the invention may be implemented using one or more general purpose processors (or special purpose processors adapted to the particular process steps and data structures) operating under program control, and that implementation of the preferred process steps and data structures described herein using such equipment would not require undue experimentation or further invention.

Inventions described herein can be used in conjunction with inventions described in International Application PCT/US99/05071, filed 8 March 1999, in the name of the same applicant, titled "Highly Available File Servers." This application is hereby incorporated by reference as if fully set forth herein. It is herein referred to as the "Availability Disclosure."

File Server System

Figure 1 shows a block diagram of a scalable and highly available file server system.

A file server system 100 includes a set of file servers 110, each including a coupled pair of file server nodes 111 having co-coupled common sets of mass storage devices 112. Each node 111 is like the file server node further described in the Availability Disclosure. Each node 111 is coupled to a common interconnect 120. Each node 111 is also coupled to a first network switch 130 and a second network switch 130.

Each node 111 is coupled to the common interconnect 120, so as to be able to transmit information between any two file servers 110. The common interconnect 120 includes a set of communication links (not shown) which are redundant in the sense that even if any single communication link fails, each node 111 can still be contacted by each other node 111.

In a preferred embodiment, the common interconnect 120 includes a NUMA (non-uniform memory access) interconnect, such as the SCI (Scalable Coherent Interconnect) interconnect operating at 1 gigabyte per second or the SCI-lite interconnect operating at 125 megabytes per second.

Each file server 110 is coupled to the first network switch 130, so as to receive and respond to file server requests transmitted therefrom. In a preferred embodiment there is also a second network switch 130, although the second network switch 130 is not required for operation of the file server system 100. Similar to the first network switch 130, each file server 110 is coupled to the second network switch 130, so as to receive and respond to file server requests transmitted therefrom.

File Server System Operation

In operation of the file server system 100, as further described herein, a sequence of file server requests arrives at the first network switch 130 or, if the second network switch 130 is present, at either the first network switch 130 or the second network switch 130. Either network switch 130 routes each file server request

in its sequence to the particular file server 110 that is associated with the particular mass storage device needed for processing the file server request.

One of the two nodes 111 at the designated file server 110 services the file server request and makes a file server response. The file server response is routed by one of the network switches 130 back to a source of the request.

Interconnect System

Figure 2A shows a block diagram of a first interconnect system for the file server system.

In a first preferred embodiment, the interconnect 120 includes a plurality of nodes 111, each of which is part of a file server 110. The nodes 111 are each disposed on a communication ring 211. Messages are transmitted between adjacent nodes 111 on each ring 211.

In this first preferred embodiment, each ring 211 comprises an SCI (Scalable Coherent Interconnect) network according to IEEE standard 1596-1992, or an SCI-lite network according to IEEE standard 1394.1. Both IEEE standard 1596-1992 and IEEE standard 1394.1 support remote memory access and DMA; the combination of these features is often called NUMA (non-uniform memory access). SCI networks operate at a data transmission rate of about 1 gigabyte per second; SCI-lite networks operate at a data transmission rate of about 125 megabytes per second.

A communication switch 212 couples adjacent rings 211. The communication switch 212 receives and transmits messages on each ring 211, and operates to bridge messages from a first ring 211 to a second ring 211. The communication switch 212 bridges those messages that are transmitted on the first ring 211 and designated for transmission to the second ring 211. A switch 212 can also be coupled directly to a file server node 110.

In this first preferred embodiment, each ring 211 has a single node 111, so as to prevent any single point of failure (such as failure of the ring 211 or its switch 212) from preventing communication to more than one node 111.

Figure 2B shows a block diagram of a second interconnect system for the file server system.

In a second preferred embodiment, the interconnect 120 includes a plurality of nodes 111, each of which is part of a file server 110. Each node 111 includes an associated network interface element 114. In a preferred embodiment, the network interface element 114 for each node 111 is like that described in the Availability Disclosure.

The network interface elements 114 are coupled using a plurality of communication links 221, each of which couples two network interface elements 114 and communicates messages therebetween.

The network interface elements 114 have sufficient communication links 221 to form a redundant communication network, so as to prevent any single point of failure (such as failure of any one network interface element 114) from preventing communication to more than one node 111.

In this second preferred embodiment, the network interface elements 114 are disposed with the communication links 221 to form a logical torus, in which each network interface element 114 is disposed on two logically orthogonal communication rings using the communication links 221.

In this second preferred embodiment, each of the logically orthogonal communication rings comprises an SCI network or an SCI-lite network, similar to the SCI network or SCI-lite network described with reference to figure 2A.

Operation Process Flow

Figure 3 shows a process flow diagram of operation of the file server system.

A method 300 is performed by the components of the file server system 100, and includes a set of flow points and process steps as described herein.

At a flow point 310, a device coupled to the file server system 100 desires to make a file system request.

At a step 311, the device transmits a file system request to a selected network switch 130 coupled to the file server system 100.

At a step 312, the network switch 130 receives the file system request. The network switch 130 determines which mass storage device the request applies to, and determines which file server 110 is coupled to that mass storage device. The network switch 130 transmits the request to that file server 110 (that is, to both of its nodes 111 in parallel), using the interconnect 120.

At a step 313, the file server 110 receives the file system request. Each node 111 at the file server 110 queues the request for processing.

At a step 314, one of the two nodes 111 at the file server 110 processes the file system request and responds thereto. The other one of the two nodes 111 at the file server 110 discards the request without further processing.

At a flow point 320, the file system request has been successfully processed.

If any single point of failure occurs between the requesting device and the mass storage device to which the file system request applies, the file server system 100 is still able to process the request and respond to the requesting device.

- If either one of the network switches 130 fails, the other network switch 130 is able to receive the file system request and transmit it to the appropriate file server 110.

- If any link in the interconnect 120 fails, the remaining links in the interconnect 120 are able to transmit the message to the appropriate file server 110.

- If either node 111 at the file server 110 fails, the other node 111 is able to process the file system request using the appropriate mass storage device. Because nodes 111 at each file server 110 are coupled in pairs, each file server 110 is highly available. Because file servers 110 are coupled together for managing collections of mass storage devices, the entire system 100 is scalable by addition of file servers 110. Thus, each cluster of file servers 110 is scalable by addition of file servers 110.

- If any one of the mass storage devices (other than the actual target of the file system request) fails, there is no effect on the ability of the other mass storage devices to respond to processing of the request, and there is no effect on either of the two nodes 111 which process requests for that mass storage device.

Alternative Embodiments

Although preferred embodiments are disclosed herein, many variations are possible which remain within the concept, scope, and spirit of the invention, and these variations would become clear to those skilled in the art after perusal of this application.

Claims

1. A file server system including

a plurality of file server nodes;

at least one inter-node connectivity element coupled to said plurality of nodes;

at least one switch coupled to said plurality of nodes and disposed for coupling file server commands to ones thereof;

said nodes including a set of pairs, each said pair being coupled to a set of storage elements and being disposed to control said storage elements in response to said file server commands.

2. A system as in claim 1, wherein at least some of said pairs are disposed for failover from a first node to a second node.

3. A system as in claim 1, wherein each said node includes a processor and a memory.

4. A system as in claim 1, wherein

each said storage element corresponds to one said pair;

each said storage element is coupled to both nodes in said corresponding pair;

whereby both nodes in said corresponding pair are equally capable of controlling said storage element.

5. A system as in claim 1, wherein said connectivity element includes a NUMA network.

6. A system as in claim 1, wherein said file server system is scalable by addition of a set of pairs of said nodes.

7. A system as in claim 1, wherein said set of storage elements coupled to at least one said pair includes a RAID storage system.

8. A system as in claim 1, wherein

each pair includes a first node and a second node;

each pair is disposed to receive file server commands directed to either said first node or to said second node;

each pair is disposed when said file server commands are directed to said first node to execute said file server commands at said first node and to store a copy of said file server commands at said second node; and

each pair is disposed when said file server commands are directed to said second node to execute said file server commands at said second node and to store a copy of said file server commands at said first node.

9. A system as in claim 8, wherein

each said pair is disposed when said file server commands are directed to said first node and said first node is inoperable to execute said file server commands at said second node; and

each pair is disposed when said file server commands are directed to said second node and said second node is inoperable to execute said file server commands at said first node.

10. A system as in claim 1, wherein

each pair is disposed to receive a file server command;

each pair is disposed so that a first node responds to said file server command while a second node records said file server command; and

each pair is disposed to failover from said first node to said second node.

11. A system as in claim 10, wherein

each pair is disposed to receive a second file server command;

each pair is disposed so that said second node responds to said second file server command while said first node records said file server command; and

each pair is disposed to failover from said first node to said second node.

12. A system as in claim 10, wherein said first node controls said storage elements in response to said file server command while said second node is coupled to said storage elements and does not control said storage elements in response to said file server command.

13. A method of operating a file server system, said method including steps for

operating a plurality of file server nodes in a set of pairs, each said pair being responsive to a set of file server commands;

coupling said file server commands to said pairs;

coupling a set of messages between ones of said nodes in a first said pair and ones of said nodes in a second said pair.

14. A method as in claim 13, including steps for failover from a first node to a second node, and from said second node to said first node, in each said pair.

15. A method as in claim 13, including steps for scaling said file server by addition of a set of pairs of said nodes.

16. A method as in claim 13, including steps for controlling a set of storage elements corresponding to one said pair from either node in said pair.

17. A method as in claim 16, including steps for operating said set of storage elements according to a RAID storage method.

18. A method as in claim 13, including steps for

receiving file server commands directed to either a first node or to a second node in each said pair;

when said file server commands are directed to said first node, responding to said file server commands at said first node and storing a copy of said file server commands at said second node; and

when said file server commands are directed to said second node, responding to said file server commands at said second node and storing a copy of said file server commands at said first node.

19. A method as in claim 18, including steps for

when said file server commands are directed to said first node and said first node is inoperable, responding to said file server commands at said second node using said copy at said second node; and

when said file server commands are directed to said second node and said second node is inoperable, responding to said file server commands at said first node using said copy at said first node.

20. A method as in claim 13, including steps for

receiving a file server command at one said pair;

responding to said file server command at a first node while recording said file server command at a second node; and

failing over from said first node to said second node.

21. A method as in claim 20, including steps for

receiving a second file server command at said one pair;

responding to said file server command at said second node while recording said file server command at said first node; and

failing over from said first node to said second node.

22. A method as in claim 20, including steps for controlling said storage elements in response to said file server command by said first node while said

second node is coupled to said storage elements and does not control said storage elements in response to said file server command.

SYSTEM 100

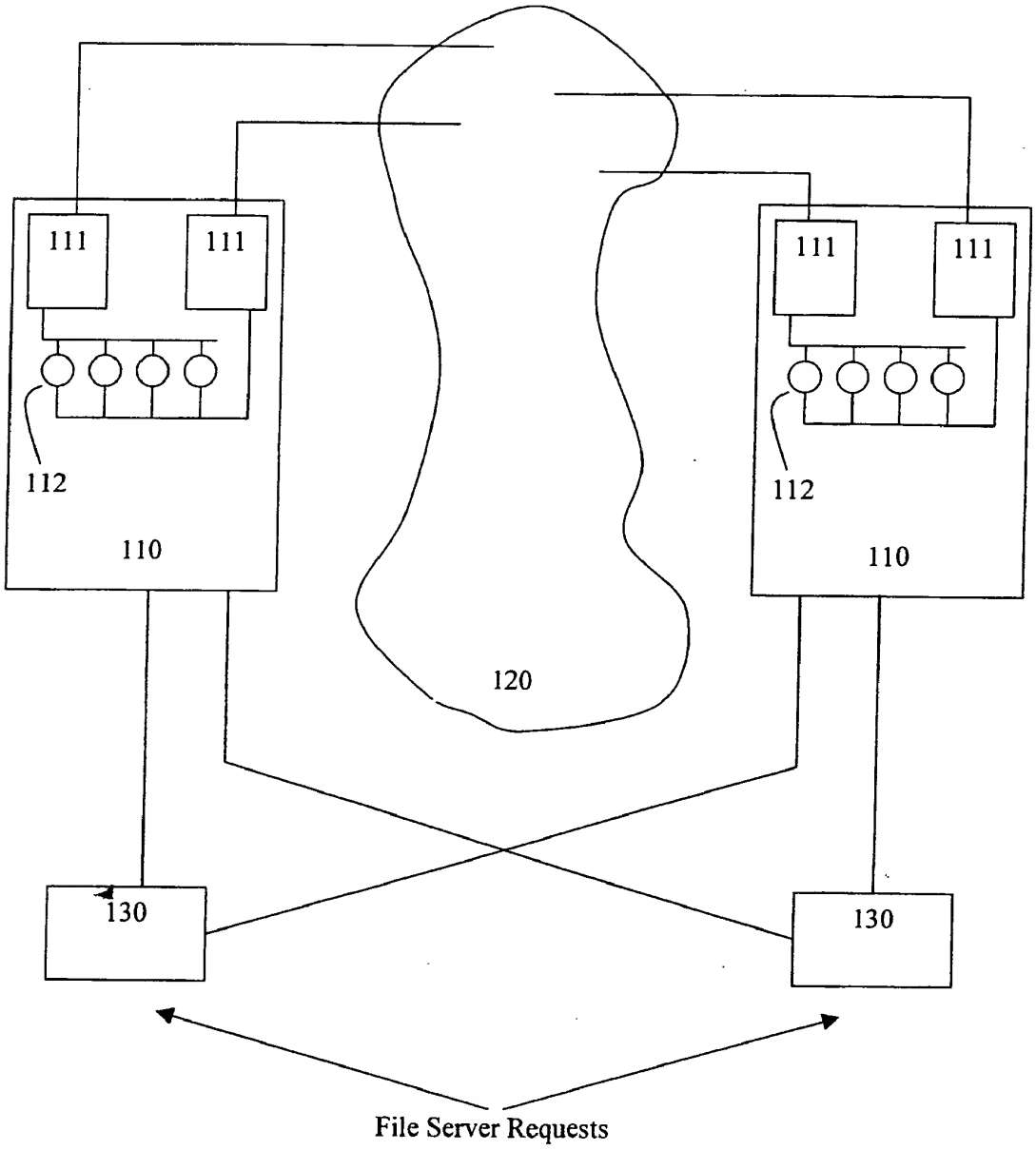


FIGURE 1

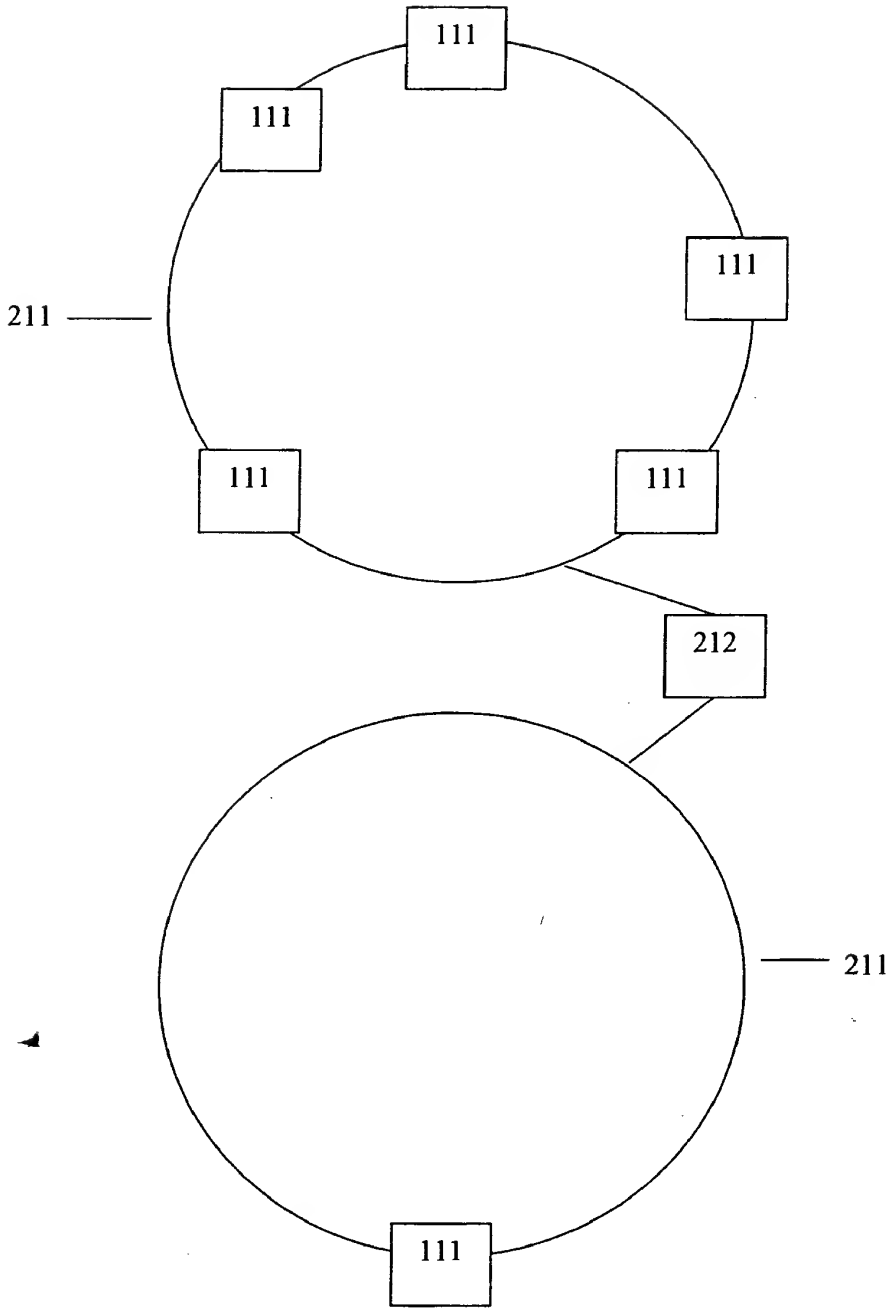


FIGURE 2A

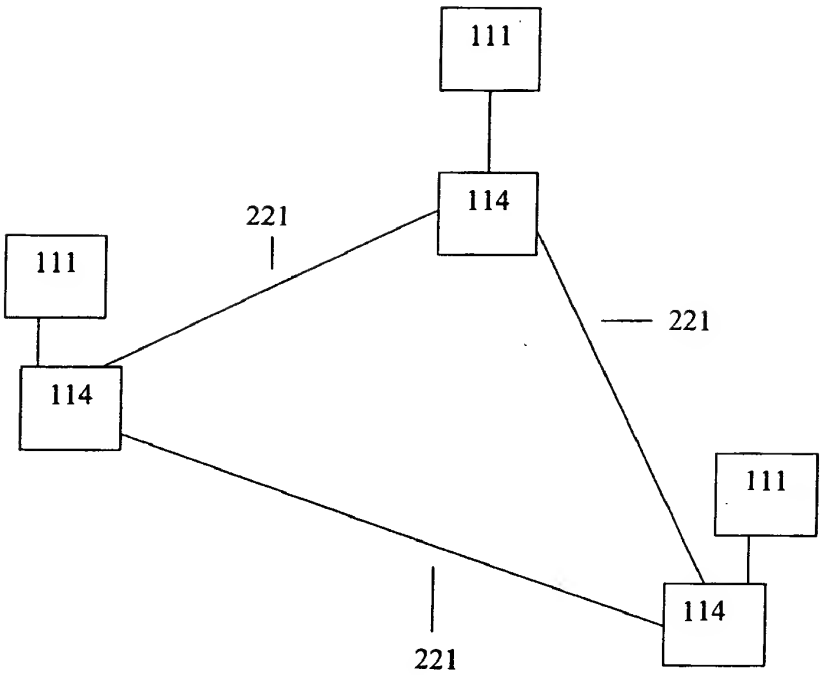


FIGURE 2B

METHOD 300

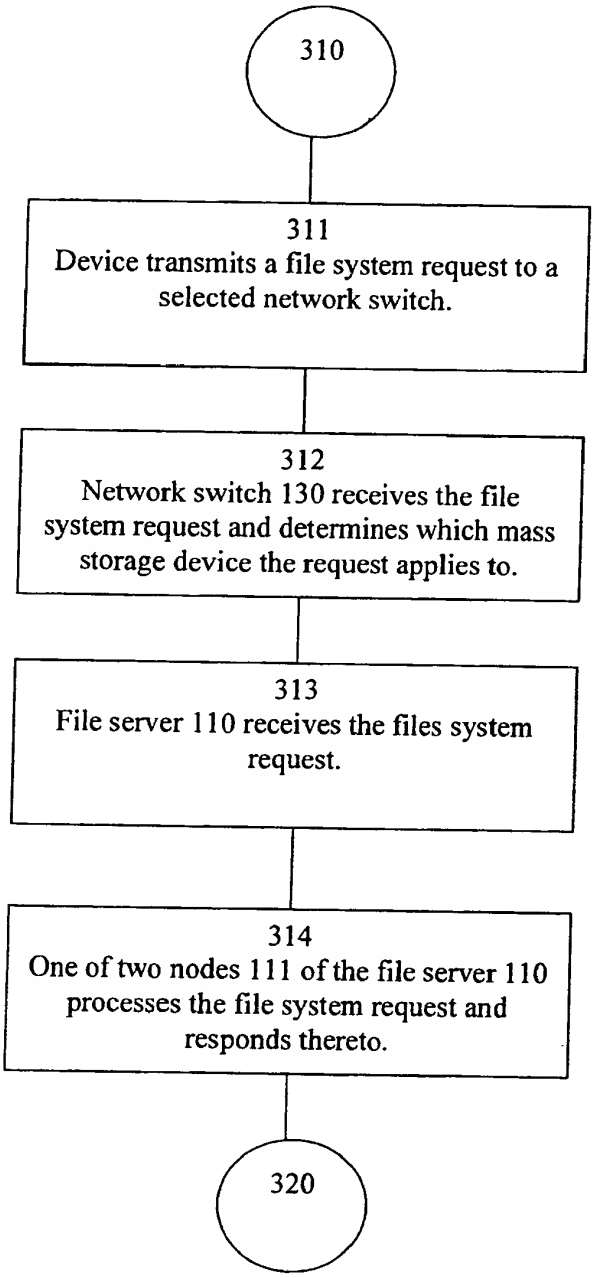


FIGURE 3